VERİMLİLİK DERGİSİ
JOURNAL OF PRODUCTIVITY

Social Navigation in Warehouse Logistics Based on Artificial Intelligence and RGB-D

Bilal Gürevin[1], Hilal Öztemel[1], Burhan Turgut Ulutürk[1], Emre Sebat[1]

ABSTRACT

Purpose: Ensuring both human safety and transportation efficiency simultaneously during the navigation of autonomous mobile robots (AMRs) in warehouse logistics is a challenging problem due to dynamic environments and diverse obstacles. In this study, a social navigation approach based on artificial intelligence was developed to optimize these two critical factors.

Methodology: RGB images from an Intel_RealSense_D455 depth camera mounted on the PIXER AMR were utilized in a YOLOv8-based model to detect humans and reach trucks (RT). For human detection, the YOLOv8 model was trained with 4746 images and labels for 362 epochs, while RT detection used 4193 images and labels for 450 epochs. Each dataset was split into 60% training, 20% testing, and 20% validation subsets. The depth feature of the camera was used to measure object distances.

Findings: Objects detected with at least 80% accuracy had their midpoints identified, and distances were calculated using the depth camera. For humans detected within 2 meters, the robot's max_speed was reduced to 80%. For RTs detected at 6 meters, a new path was planned.

Originality: This study provides a novel integration of social navigation and deep learning to address the dual challenge of ensuring safety and efficiency in AMR navigation, contributing to advancements in warehouse logistics.

Keywords*:* Artificial Intelligence, Social Navigation, Warehouse Logistic.
JEL Codes: C63, L62, R41.

Yapay Zekâ ve RGB-D Tabanlı Depo İçi Lojistikte Sosyal Navigasyon

ÖZET

Amaç: Depo lojistiğinde otonom mobil robotların (AMR) navigasyonu sırasında insan güvenliğini ve taşıma verimliliğini aynı anda sağlamak, dinamik ortamlar ve çeşitli engeller nedeniyle zor bir problemdir. *Bu* çalışmada, bu iki kritik faktörü optimize etmek amacıyla yapay zekâ tabanlı bir sosyal navigasyon yaklaşımı geliştirilmiştir.

Yöntem: PIXER AMR üzerine monte edilmiş Intel_RealSense_D455 derinlik kamerasından alınan RGB görüntüler, insan ve Reach Truck (RT) algılaması için YOLOv8 tabanlı bir modelde kullanılmıştır. İnsan algılama için YOLOv8 modeli 4746 görüntü ve etiketle 362 epoch boyunca, RT algılama için ise 4193 görüntü ve etiketle 450 epoch boyunca eğitilmiştir. Her bir veri seti, %60 eğitim, %20 test ve %20 doğrulama alt kümelerine bölünmüştür. Algılanan nesnelerin mesafeleri, kameranın derinlik özelliği kullanılarak ölçülmüştür.

Bulgular: En az %80 doğrulukla algılanan nesnelerin orta noktaları belirlenmiş ve mesafeleri derinlik kamerası kullanılarak hesaplanmıştır. İnsanlar 2 metre mesafede algılandığında robotun maksimum hızı %80'e azaltılmıştır. RT'ler 6 metre mesafede algılandığında yeni bir rota planlanmıştır.

Özgünlük: Bu çalışma, AMR navigasyonunda güvenliği ve verimliliği sağlama ikili problemini ele alan sosyal navigasyon ve derin öğrenmenin yenilikçi bir entegrasyonunu sunarak depo lojistiğinde ilerlemelere katkı sağlamaktadır.

Anahtar Kelimeler*:* Yapay Zekâ, Sosyal Navigasyon, Depo Lojistiği.
JEL Kodları: C63, L62, R41.

[1] Ottobo Robotics and Artificial Intelligence Technologies Inc., İstanbul, Türkiye

Corresponding Author: Bilal Gürevin, bilalgurevin@gmail.com
DOI: 10.51551/verimlilik.1523828
Research Article | Submitted: 30.07.2024 | Accepted: 30.12.2024
Cite: Gürevin, B., Öztemel, H., Ulutürk, B.T. and Sebat, E. (2025). "Social Navigation in Warehouse Logistics Based on Artificial Intelligence and RGB-D", *Verimlilik Dergisi,* 59 (2), 301-322.

1. INTRODUCTION

With the emergence of Industry 4.0, both academic and industrial studies on autonomous mobile robots (AMR) have been increasing. The increasing use of AMR in warehouse logistics has provided significant advances in operational efficiency and safety protocols (Gürevin et al., 2024; Fragapane et al., 2022). In response to the increasing demands of fast delivery services and e-commerce operations, the integration of AMRs into warehouse environments has become inevitable. These robots are designed to navigate dynamic environments, interact seamlessly with human employees, and ensure efficient transportation of products. However, maintaining transportation efficiency while ensuring human safety poses a major challenge (Trakadas et al., 2020; Cognominal et al., 2021). When looking at the literature, artificial intelligence-based object recognition systems have been developed in warehouse environments thanks to camera sensors on robots. However, these studies are mostly in areas where personnel are checked to see if they are wearing safety equipment, human detection is performed, or autonomous personnel tracking systems are performed. It does not directly affect AMR navigation in terms of efficiency.

RT vehicle and human are indispensable personnel in many warehouse environments. It is important for AMRs to take these two objects, which are constantly dynamic, into consideration in terms of both occupational safety and process efficiency. The situations that AMRs should pay attention to are as follows; reach trucks (RT) can cause route congestion due to their large structures and cumbersome movements. This has a negative impact on the AMR working process. On the other hand, humans should be considered as a constantly moving entity that requires attention. In this study, it is aimed to develop a social navigation system that increases both human safety and transportation efficiency in warehouse environments. It is aimed to provide a harmonious interaction between robots and their environment by using advanced sensor technologies such as depth cameras (Gürevin et al., 2023) and the latest deep learning algorithms.

The central part of our method is the use of the Intel RealSense D455 depth camera, which provides object detection and distance measurement by capturing RGB images and depth data. Object detection and distance measurement operations were performed using the images taken from this depth camera in front of our robot, which we named PIXER and developed by us (Ottobo Robotics and Artificial Intelligence Technologies Inc.). The YOLOv8 model was used for human and RT detection. In the training of each model, the epoch values were set to 500. However, the save_period value was set to “-1” and the patience value was set to 10. In this way, the training continued until the last 10 epochs, where the verification losses did not improve. For this reason, each model could not continue its training until 500 epochs. The human detection model was trained using the transfer learning method with 60% training, 20% test and 20% validation data ratios for 362 epochs on a dataset containing 4746 images and labels. Similarly, the RT detection model was trained using the same data distribution for 450 epochs using transfer learning with 4193 images and labels. Our system ensures safety by reducing the robot's maximum speed to 80% when a human is detected at a distance of 2 meters on the robot navigation route. During navigation, when an RT is detected at a distance of 6 m on the route in narrow corridors, an alternative route is planned and the robot's path is optimized. The novelty of this research lies in its dual-focused approach to improve both safety and operational efficiency through social navigation. By integrating deep learning-based object detection and real-time distance measurement, our system provides a robust solution for the evolving needs of warehouse logistics.

The remainder of the paper is structured as follows; Section 2 reviews the existing literature on AMR navigation and safety systems. Section 3 details the methodology, including data collection, model training, and system implementation. Section 4 presents the experimental results and their analysis. Finally, Section 5 concludes the study and discusses future research directions.

2. LITERATURE RESEARCH

In this section, the existing studies on AMRs in the fields of human-robot interaction and social navigation were comprehensively reviewed. In addition, the potential of AMRs to increase operational efficiency and safety, the challenges encountered in achieving these goals, and the innovative solutions developed to overcome these challenges were discussed. This literature review aims to reinforce the scientific basis of our research by allowing us to understand the previous research on which our study is based and the methods used. The studies obtained from the literature are as follows;

Zhu and Zhang conducted a research article on deep reinforcement learning-based navigation studies. In their deep learning-based social navigation studies, they provided the generation of speed and volumetric (radius) estimates of pedestrians with point cloud data taken from lidar and depth cameras (Zhu and Zhang, 2021). Francis and his colleagues determined the benchmark criteria for the evaluation of social navigation studies conducted by researchers with their comprehensive study. They also defined the social navigation robot (Francis et al., 2023). Daza and his colleagues proposed a navigation algorithm that takes into account the interaction of a robot against humans and robots, unlike robot-robot or human-robot interaction. They analyzed the proximity of people to each other and the navigation of other robots and provided the planning of the robot's behavior. They modeled their work in the Gazebo simulation environment (Daza et al., 2021). The performances of various YOLO, SSD, RCNN, R-FCN and SqueezeDet applications were evaluated. YOLO v3-416 emerged as an ideal model for embedded platforms by providing relatively accurate results in a reasonable time (Kim et al., 2018).

Mayershofer and his colleagues presented the Logistics Objects in COntext (LOCO) dataset, which is the first publicly available object recognition dataset in the logistics field. LOCO contains 39,101 images, and the first version includes 5,593 bounding-box labeled images. A total of 151,428 pallets, small load carriers, fixed racks, forklifts and pallet trucks were labeled (Mayershofer et al., 2020). Salmerón-García et al. (2015) investigated the effects of cloud computing on robotic navigation and demonstrated the advantages of performing vision-based navigation tasks over the cloud. In their work, they used a mobile robot based on the Robot Operating System (ROS) and processed the information received from stereo cameras on the robot on a cloud platform consisting of five bare-metal nodes. (Salmerón-Gar Salmerón-García et al., 2015).

Kenk and his colleagues addressed the reliable navigation capabilities required by industrial and mobile robots to provide safe and comfortable navigation in environments full of people. As a result of their work, they were able to ensure that mobile robots could approach pallets to pick up objects while maintaining a certain distance from people (Kenk et al., 2019). Wang and his colleagues, starting from the lack of public datasets that can be used for object detection in a warehouse environment, collected a large number of images in a real warehouse environment and marked them with cameras. This study enabled the accurate detection of warehouse objects (Wang and Li, 2023).

Clavero et al. (2024) presented a new architecture called DMZoomNet in the intralogistics industry. This architecture aims to improve object detection performance by combining deep learning-based detectors with distance information. In their work, they evaluated their method on the LOCO dataset, one of the few open-source datasets specifically designed for intralogistics scenarios (Clavero et al., 2024). In the study proposed by Truong and Ngo (2016), an effective human comfort safety framework was proposed for mobile service robots to navigate safely and socially in social environments. The dynamic social zone (DSZ) based human comfort safety framework enabled the robot to estimate the target position when approaching a human or group of humans, allowing the robot to both avoid and approach humans in a socially acceptable manner (Truong and Ngo, 2016).

Mavrogiannis et al. (2023), in their review article, organized a large set of open issues related to robot planning, behavior design, and evaluation methodologies to address the challenges of robot navigation in crowded environments (Mavrogiannis, 2023). Beyer et al., worked on improving the ability of mobile robots to detect people, especially those using mobility aids. For this purpose, they developed DROW, a deep learning-based wheelchair and walker detector (Beyer et al., 2018). Jia et al. proposed a person detection network that uses an alternative strategy to combine scans obtained at different times. Their work outperformed the current state-of-the-art methods on the DROW dataset and was approximately four times faster (Jia et al., 2020).

When the current literature is examined, it is seen that significant progress has been made in human detection and social navigation with various approaches and technologies. However, there is a lack of a comprehensive approach in the literature that combines human detection and social navigation, takes into account both human and mobile obstacles such as RT, and prioritizes safety and efficiency. The studies conducted were generally conducted by simulating them in a theoretical virtual environment. Or they were limited to object detection only. In this study, a new system has been proposed that detects both humans and RT vehicles using the deep learning-based YOLOv8 model and depth camera, and provides integration with social navigation strategies. Thanks to this system, the robot in navigation will be instantly made sensitive to dynamic objects such as humans and RT in its environment. This innovative approach aims to go beyond existing systems and ensure the safe and effective operation of autonomous mobile robots in e-commerce warehouses.

3. METHOD

In this section, the methods used to develop the social navigation system in the PIXER robot were presented in detail. In the study, Intel RealSense D455 depth camera (Intel Realsense, 2024) and YOLOv8 deep learning framework (Ultralytics, 2024a) were used for human and RT detection. In the images taken from the depth camera positioned in front of the robot, which we call PIXER, human and RT were detected and their midpoints were determined. Thanks to the depth feature of the camera, the distance of this point to the robot was measured instantly. Thus, the human and RT distances on the robot's navigation route were measured and the robot was enabled to take the necessary actions. In case a human was detected within 2 m. on the robot's route, the robot's maximum speed was reduced to 80%. Again, in the RT detection of the robot in narrow corridors and in the case where RT restricts the robot's movement, the robot was provided with a new alternative path planning after a 5-second waiting period. The representative flow diagram of the structure mentioned is as given in Figure 1.

In the following subsections, the dataset used to obtain the artificial intelligence model developed for human and RT detection, model training, object detection and distance measurement, and finally system integration were discussed.

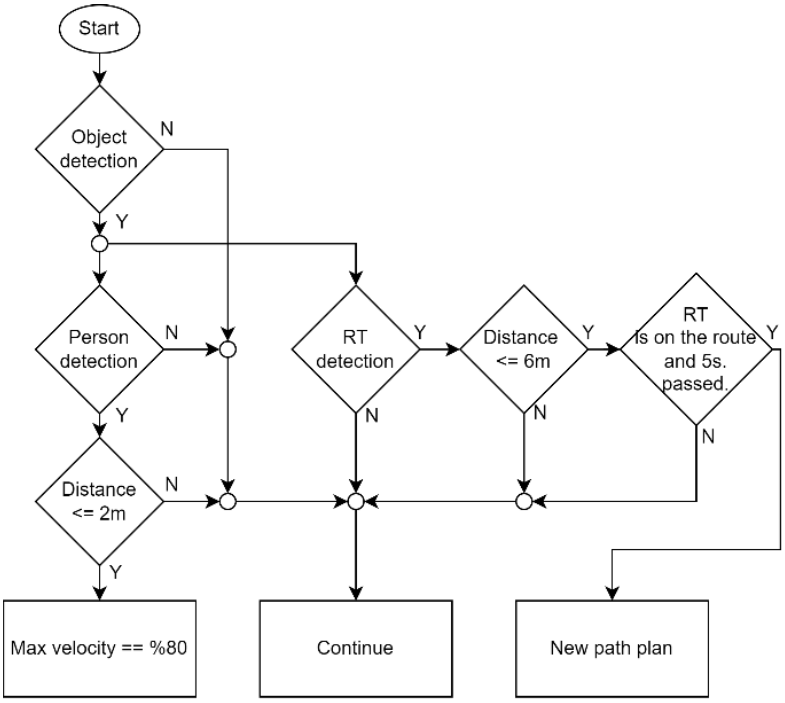


Figure 1. Social navigation scenario for PIXER AMR

3.1. YOLOv8 Deep Learning Model

YOLOv8 algorithm is one of the latest deep learning algorithms for object detection introduced by Ultralytics. YOLOv8 is optimized for real-time object detection and offers various model sizes; YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l and YOLOv8x. This flexibility allows the model to be used in a wide range of applications with different computational resources. YOLOv8 has the same architecture as previous versions, but offers several improvements compared to previous YOLO versions. It integrates both “Feature Pyramid Network” (FPN) and “Path Aggregation Network” (PAN) using a new neural network architecture. The FPN is an architecture that facilitates multi-scale object detection by efficiently transferring information across different layers of a deep neural network. FPN combines high-resolution, low-level features from earlier layers with low-resolution, high-level features from deeper layers. This enables the model to detect both small and large objects simultaneously. FPN primarily provides an upward flow of information, optimizing feature propagation to better interpret objects of various sizes. The PAN, on the other hand, complements the upward information flow of FPN by introducing a downward flow of information. PAN allows high-level abstract features to be aggregated with lower-level detailed features, creating a bidirectional information flow. This enhances detection accuracy and improves the model’s ability to detect objects at different scales, especially in complex scenes. PAN is specifically designed to bolster the success of multi-scale object detection by refining feature aggregation. The architecture of the YOLOv8 model is given in Figure 2 (Reis et al., 2023). In this study, the YOLOv8n artificial intelligence model was used to perform social navigation of AMRs actively working in logistics warehouses. The PIXER robot, which is actively used in the warehouse environment, has power-consuming industrial sensor systems such as a 2D LIDAR, a 3D LIDAR and a depth camera. For this reason, the YOLOv8n model was preferred because it is a lightweight and fast model that requires less computational resources.

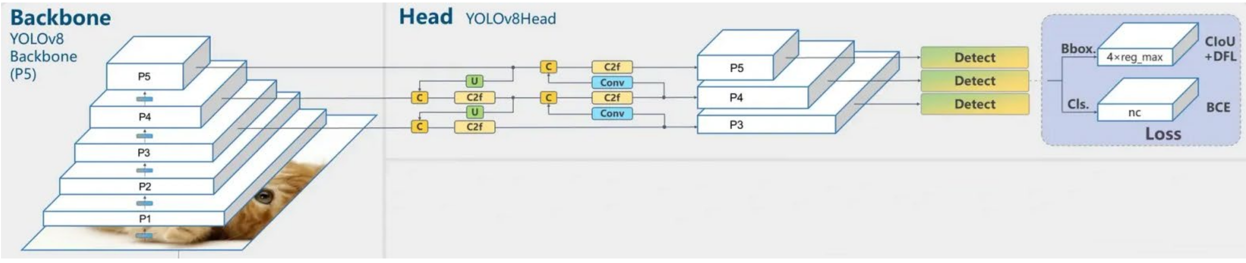


Figure 2. YOLOv8 architecture

3.2. Data Set

The dataset used in this study was prepared in two separate categories, for human and RT detection. A dataset consisting of 4746 images and their corresponding labels was created for human detection. This dataset was divided into three groups as 60% training, 20% test and 20% validation. 4193 images were used for RT detection. 1% of the images were used as background data. The reason for keeping the background data in a small proportion is that this type of data is required in less numbers compared to the object to be detected. Because the model does not need a large number of examples to learn the background. 1% background data is sufficient to increase the overall performance of the model. Using too much may negatively affect the detection success or lead to unbalanced results. This dataset was also divided into training, test and validation groups in the same proportions. The dataset distribution is as shown in Figure 3.

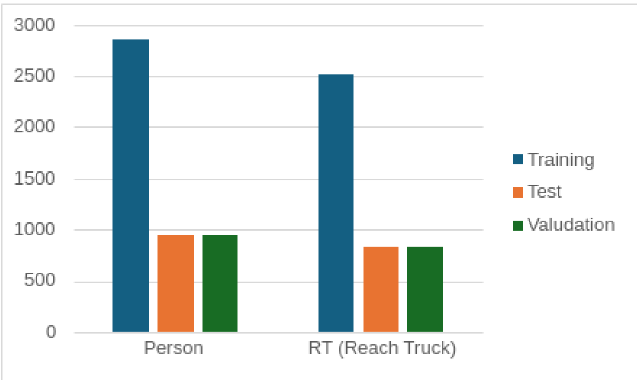


Figure 3. Dataset distribution

In creating the datasets, images taken in storage environments were used. In addition, data augmentation techniques were used to ensure that the datasets were balanced and diverse. These images were obtained from various lighting, angles and positions to reflect real-world conditions. These techniques include operations such as rotation, scaling, brightness and contrast adjustments. Some of the human and RT images obtained with data augmentation are given in Figure 4.

Figure 4. RT and human images obtained by data augmentation method for model training.

The human dataset contains a single class, ‘person’. On the other hand, the RT direction dataset contains three classes, ‘Reachtruck Rear’, ‘Reachtruck Sides’, and ‘Reachtruck Front’. This distinction was made because the gaze direction of the RT that PIXER will encounter during its movement in the field needs to be evaluated. By measuring the gaze direction of the RT and its distance from the robot, it will be possible to comment on the direction in which it is moving.

3.3. Model Training

The dataset of the model in which the YOLOv8n artificial intelligence model developed for RT and human detection was used and the model was trained is as described in the template in Figure 5. In order to provide high detection accuracy for both objects and to facilitate the maintenance and development processes of the model in R&D studies, training was performed on two different models. The YOLOv8n model used for human detection was trained for 362 epochs. The model was trained for RT direction detection for 450 epochs. During the training process, hyperparameter optimization was performed to increase model performance. The accuracy of the model was measured on the test and validation sets, and 75% accuracy was achieved for human detection and 93% for RT direction detection.

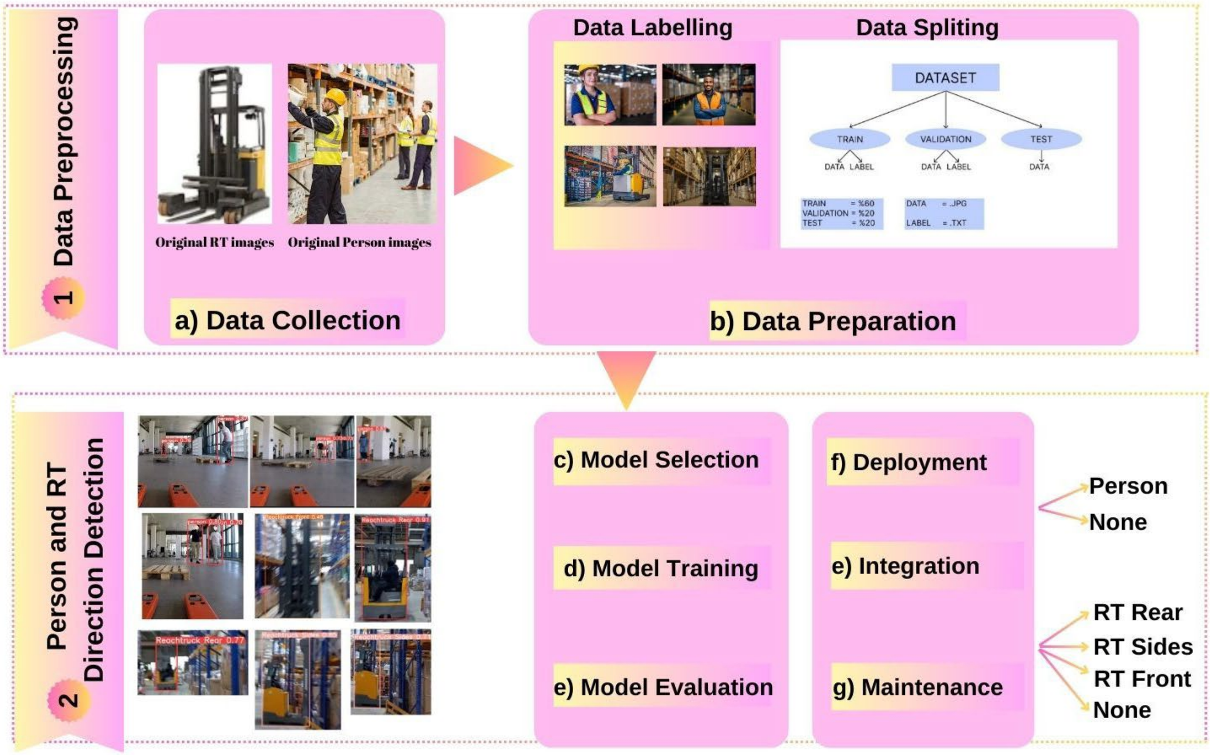


Figure 5. Human and RT detection model methodology

During training, models like YOLO use Focal Loss for DFL Loss, Complete Intersection over Union (CIoU) for Box Loss and Binary Cross-Entropy (BCE) for Cls Loss as loss functions (Ultralytics, 2024b). The optimization process was performed with Adam optimizer for human detection model and Stochastic Gradient Descent (SGD) for reach truck direction detection. For SGD optimizer, learning rate was set as lr0=0.01 and momentum was set as 0.9, weight decay was set as 0.0 and 0.0005 for different layers. For Adam optimizer, learning rate was set as lr0=0.01 and momentum was set as 0.9, weight decay was set as 0.0 and 0.0005 for different layers. Automatic Mixed Precision (AMP) was used in the training process to increase speed and memory efficiency. The image size was set to 640, the scaling factor was set to 0.5, the initial and final learning rates were set to lr0 and lrf to 0.01. The training period was configured as 1000 epochs, but the early stopping parameter was set to 50, and the RT direction detection training was completed in 450 epochs.

During training, the YOLOv8n model was trained by applying transfer learning to both models. Early stopping with a patience value of 50 was used. YOLO's early stopping mechanism stops training when there is no improvement in the validation loss or validation metric (mAP) for a certain period of time (50 epochs) during model training. This method is used to prevent overfitting and to save time and resources. When the model's validation error does not improve, the training process is automatically stopped and the training is terminated at the point where the model shows the best performance. In YOLO models, training is usually structured with a high number of epochs (e.g. 1000), but thanks to early stopping, training is stopped at the point where the performance peaks. This prevents training for unnecessary epochs and ensures efficient use of resources.

Validation loss based early stopping is a frequently used technique in YOLO models. During model training, the loss on the validation set is monitored and if the validation loss does not improve over several epochs, the training process is stopped. This prevents the model from being unnecessarily overtrained and overlearning, and ensures that the model with the best performance is preserved. In the data augmentation technique; HSV values were set as hsv_h: 0.015, hsv_s: 0.7, hsv_v: 0.4, 5 degree rotation and 0.1 translation are applied to the images. Batch size was set as 16, and the maximum number of objects that the model can detect was set as max_det=300. The save_period parameter was set as -1, and the model with the highest accuracy was saved. In YOLO models, cross-validation methods are generally not used. Instead, training and validation sets are used to evaluate the generalization ability of the model. In this study, the performance of the model was evaluated using the validation data set.

The use of GPU is of great importance to accelerate the training process. For this reason, training was performed using Amazon servers with CUDA and CuDNN features. The AMI (Amazon Machine Image) used includes the Ubuntu 20.04.6 LTS (GNU/Linux 5.15.0-1048-aws x86_64) operating system, 535.104.12 NVIDIA driver version, NVIDIA A10G GPU and 12.1 CUDA version.

3.3.1. Model Performance

Commonly used metrics to evaluate the performance of object detection models include Precision, Mean Average Precision (mAP), Recall and F1. These metrics measure the ability of the model to correctly identify objects. Precision is the ratio of true positives to total predictions. Recall is the ratio of true positives to the total number of positive examples. F1 score is the harmonic mean of precision and recall. MAP (Mean Average Precision) measures the overall accuracy of the model. MAP50 and MAP50-95 express the precision of the model in certain correct prediction intervals. These metrics are used to comprehensively evaluate the performance of the model in object detection tasks (Önal and Dandıl, 2021). The performance evaluations of the models developed in this study were given in Table 1. In addition, a detailed review was given in Section 4.

Table 1. Performances of RT and human detection models

Model Name	*Precision (%)*	*Recall (%)*	*F1 (%)*	*mAP50 (%)*	*mAP50-95 (%)*
Person detection model	95	97	97	97	75
RT detection model	98	99	99	99	93

The main objective of this project is to obtain maximum efficiency with minimum data. That's why, transfer learning (YOLOv8n), data augmentation and error-driven learning techniques, which is one of the active learning strategies, were used. The model was developed in a structure that can learn from its errors with the principle of error-driven learning. In the first stage, the model was trained and tested with 500 images. According to the test results, the erroneously detected data were corrected and added to the training data set, thus ensuring that the model learned from errors. The model was retrained with the updated data set and its performance gradually improved. This process was repeated in order to continuously improve the performance of the model. In order to prevent overfitting, especially in small data sets, the early stopping parameter was set to 50 and the dropout rate was applied as 0.5. In this way, the model was able to generalize by increasing its overall performance despite being trained with limited data. In addition, occlusion tests of the model in difficult scenarios were also performed. The results of these tests are shown in Figure 6. The model was evaluated with its ability to detect only partially visible objects during occlusion tests. In these tests, situations where objects were covered to a certain extent and were not clear were simulated. Especially thanks to YOLOv8's bounding box strategy, the model managed to make correct predictions by evaluating only the visible parts. We would also like to point out that the early stopping and dropout parameters were optimized to reduce the risk of overfitting against different scenarios. Thanks to this, the model was able to generalize even on small data sets.

Figure 6. Occlusion test results for the RT detection model

YOLOv8's bounding box strategy is a method developed to increase the precision of the location and size of objects and makes object detection more flexible. This strategy works by dividing a bounding box into 9 small parts in order to detect objects more precisely using a grid-based system. This strategy analyzes different parts of the object separately by dividing each bounding box into a 3x3 grid. In this way, the model can accurately detect the object even in difficult scenarios such as occlusion by using only the visible parts of the object. Especially in difficult scenarios such as occlusion (covering a part of the object) and changing light conditions, it enables detection by only using the visible parts of the objects. Thanks to the division of the bounding boxes into small parts, the model can make more accurate predictions by analyzing different parts of the object and shows high performance in these difficult conditions. This approach provides better capture of small objects and strengthens contextual information. Thus, the overall accuracy of the model increases and reduces the risk of overfitting.

3.4. Object Detection and Distance Measurement

In this section of the study, object detection and distance measurement methods were detailed. The images obtained with the RT and human detection models developed using YOLOv8n were enclosed in a bounding box around the objects. In order for the RGB and depth images to be compatible, the alignment process was performed with the Intel RealSense D455 depth camera, and thus the pixel and position values from both cameras (depth and RGB) were matched. Otherwise, although the images taken from the depth belt and the RGB camera have the same frame sizes, they have different depth image outputs. The alignment situation in question was expressed in the images in Figure 7 and Figure 8.

In order to increase the accuracy of distance measurement in dynamic environments, sensor noise for RealSense D455's depth data was reduced by various filtering techniques. In this context, "spatial", "temporal" and "hole filling" filters were used while obtaining depth data. Spatial filter was used to reduce sudden jumps in distance data by considering the relationship of each pixel with its neighbors. Temporal filter was used to ensure temporal accuracy between multiple consecutive frames. Hole filling filtering was used to fill the gaps that the camera has difficulty detecting. Thanks to these filtering processes, sensor noise that may occur especially while the robot is moving was minimized.

The midpoints of the detected objects were determined and the distances of these midpoints to the robot were calculated using the depth data provided by the Intel RealSense D455 depth camera. In this way, the distance information of the detected objects was transferred to the ROS (Robot Operating System) environment and the control of the PIXER robot was provided. The flow diagram explaining the object detection and distance measurement was given in Figure 9.

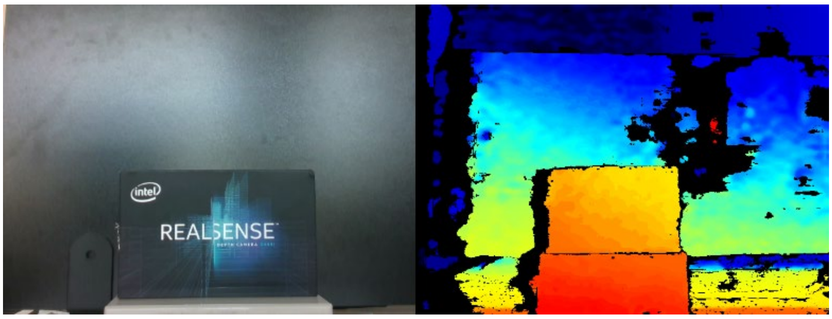


Figure 7. RGB (left) and depth image (right) without alignment and filtering operations

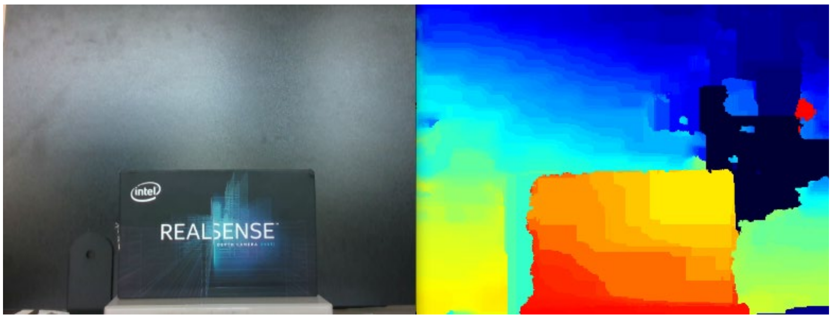


Figure 8. RGB (left) and depth image (right) with alignment and filtering operations

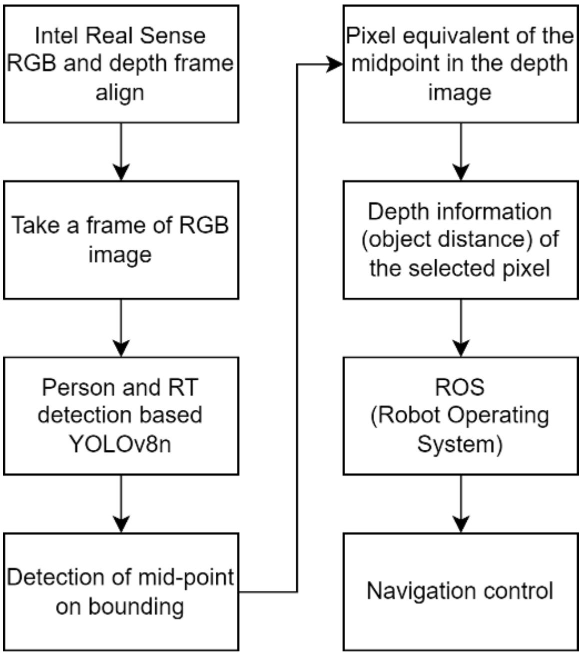


Figure 9. Flowchart of integration of object detection and distance measurement with RGB-D into PIXER navigation

3.5. PIXER Robot

The studies carried out in this research were carried out on the industrial AMR developed by us and named PIXER. The PIXER robot, which works in logistics companies, has a carrying capacity of 80 kg and is used for autonomous product collection based on ROS. Working in large warehouses such as 10000 m2, PIXER does not require any extra markers while performing the navigation process with the 3D LIDAR located on the tower (top of the robot). In addition, it ensures safety during navigation the 2D LIDAR located at the bottom. The features of the PIXER robot are as given in Table 2. In addition, images of the PIXER robot are given in Figure 10. To ensure the safety of the robot during navigation; Velodyne 3D LIDAR, RealSense Depth Cam and SICK 2D LIDAR sensors wer used. The work presented in this article was designed as an additional protection layer to the existing sensors. Even if there are cases where the detection processes are faulty, this protection infrastructure prevents accident risks.

Table 2. PIXER AMR features

Features	*Sensor and Capacity*
Navigation	3D - Velodyne LIDAR
Security	2D - Sıck LIDAR
Carrying Capacity	80 kg
Battery	24 V - 44 Ah (Wireless)
Camera	İntel Real Sense 455
PC	Intel i7, 11. Gen, 16 Gb, 256 SSD

Figure 10. PIXER AMR robot

4. EXPERIMENTS and RESULTS

In this study, a YOLOv8n-based artificial intelligence model was trained for the purpose of detecting humans and RT vehicles. Awareness of humans and RT vehicles was achieved during the robot's navigation by utilizing a pretrained model trained with the YOLO algorithm.

The training was conducted on an AMI (Amazon Machine Image) machine equipped with an NVIDIA A10G GPU and CUDA version 12.1. The pixel value corresponding to the midpoint of detected objects in the RGB image was also marked in the depth image, enabling distance measurement based on this point. Object detection was performed by running the YOLOv8n model on each frame of the RGB images. The central coordinates of detected objects, along with their depth information, were utilized to inform the robot's movement strategies. In social navigation strategies, when a human was detected within 2 meters in front of the robot, the maximum speed was reduced by 80%. Similarly, if an RT vehicle was detected within 6 meters along the path, an alternative route was generated to plan a new path. These strategies were developed to ensure human safety while enhancing operational efficiency. The following sections include performance evaluations of the model used and social navigation environment tests.

4.1. Model Training Evaluation

In this part, the results reached from experimental test were presented. Object detections accuracy, to measure distance accuracy and were evaluated for general performance about social navigation system. The performance of the person and RT detection model developed with YOLOv8n was assessed. Otherwise, this model was integrated on the PIXER robot and was tested in a warehouse environment.

4.1.1. Human Detection Model Performance

A confusion matrix is a useful tool for evaluating the performance of a classification model by comparing predicted values to actual values. Figure 11 shows to evaluated model performance about person model.

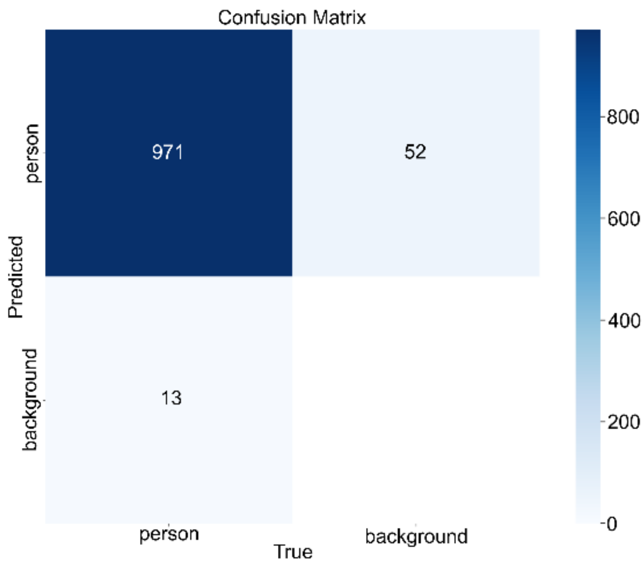


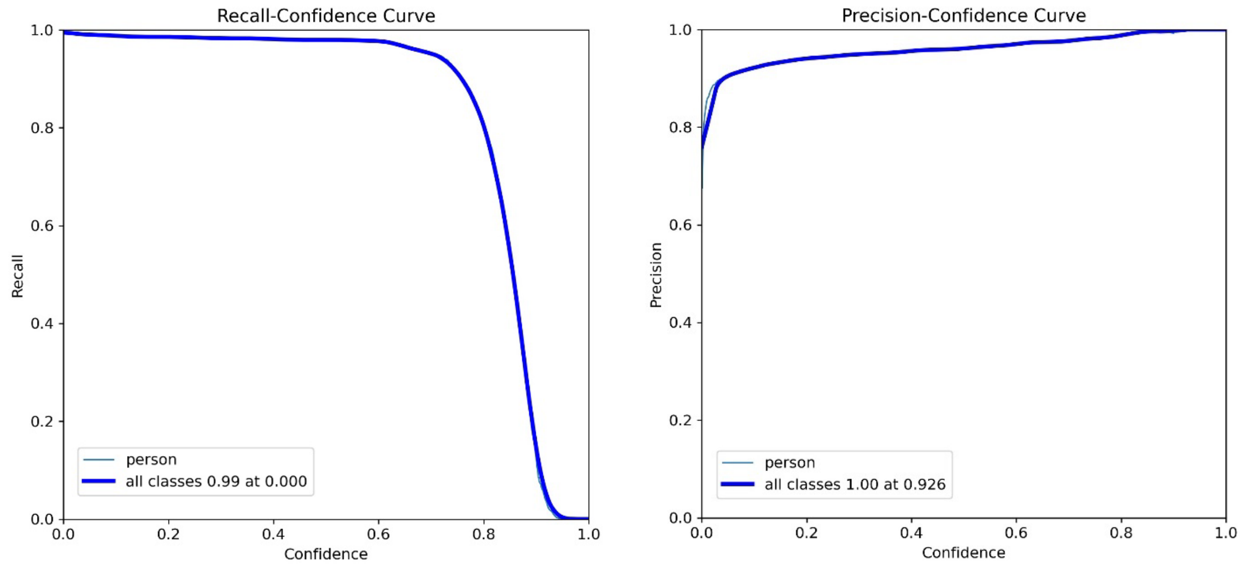
Figure 11. Confusion matrix of human detection model

The evaluation of the matrix is as follows:

- True Positives (TP) were shown as 971 (in the top left cell). This indicates that the model correctly predicted the "person" class 971 times.
- False Positives (FP) were recorded as 52 (in the top right cell). This means that the model incorrectly predicted the "background" class as "person" 52 times.
- False Negatives (FN) were determined to be 13 (in the bottom left cell), which means that the model incorrectly predicted the "person" class as "background" 13 times.
- True Negatives (TN) were not shown in the bottom right cell. This indicates that the model either could not accurately identify the "background" class or completely ignored it.

The color scale represents the magnitude of the values in the matrix cells; dark blue corresponds to high values, while light blue corresponds to low values. Overall, the high number of TPs indicates that the model is successful in correctly identifying the "person" class. However, the FP and FN values reveal that the model makes some errors. The lack of TN indicates that the model has struggled to predict the "background" class. During model training, data belonging to the "background" class was not utilized. It was deemed unnecessary to include background data as the model (Yolov8n), having already developed its object detection capabilities through transfer learning, is focused on detecting the "person" class. Therefore, the accuracy of the background class is not critically important in this application scenario.

In Figure 12(a), the statement “all classes 0.99 at 0.000” in the recall score graph indicates that the recall scores for all classes are 99% when a threshold value of 0 is used. This suggests that the model is able to successfully recall all classes even at a very low threshold value. Additionally, the statement “all classes 1 at 0.926” in the precision score graph in Figure 12(b) indicates that the precision scores for all classes are 100% when a threshold value of 0.926 is used. This demonstrates that the model has an excellent ability to correctly classify all classes when decisions have made above the 0.926 threshold.



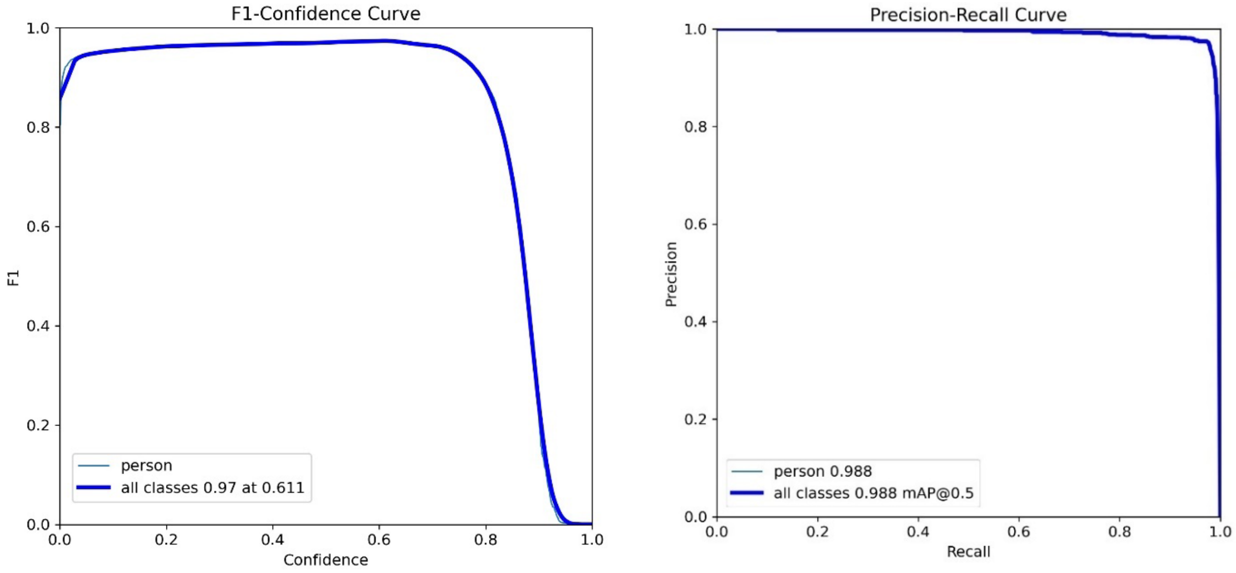
(a) (b)

Figure 12. Recall-Confidence and Precision-Confidence score graph of human detection model

In Figure 12(a), the statement “all classes 0.99 at 0.000” in the recall score graph indicates that the recall scores for all classes were 99% when a threshold value of 0 was used. This suggests that the model is able to successfully recall all classes even at a very low threshold value. Additionally, the statement “all classes 1 at 0.926” in the precision score graph in Figure 12(b) indicates that the precision scores for all classes are 100% when a threshold value of 0.926 is used. This demonstrates that the model has an excellent ability to correctly classify all classes when decisions have made above the 0.926 threshold.

In Figure 13(a), the statement “all classes 0.97 at 0.611” in the f1 score graph indicates that the all f1 scores for all classes were 97% when a threshold value of 0.611 was used. This suggests that the model is able to successfully f1 all classes even at a very high threshold value. Additionally, the statement “all classes 0.988 at mAP@50” in the precision-recall score graph in Figure 13(b) indicates that the mAP scores for all classes are 98%. This suggests that the model is able to successfully recall and precission all classes even at a very low threshold value.

In Figure 14, it displays the losses and performance metrics of the object detection model during the training and validation process. Each of these graphs allows us to analyze a specific aspect of the model and its performance in detail.



(a) (b)

Figure 13. F1 and Precision-Recall score graph of the human detection model

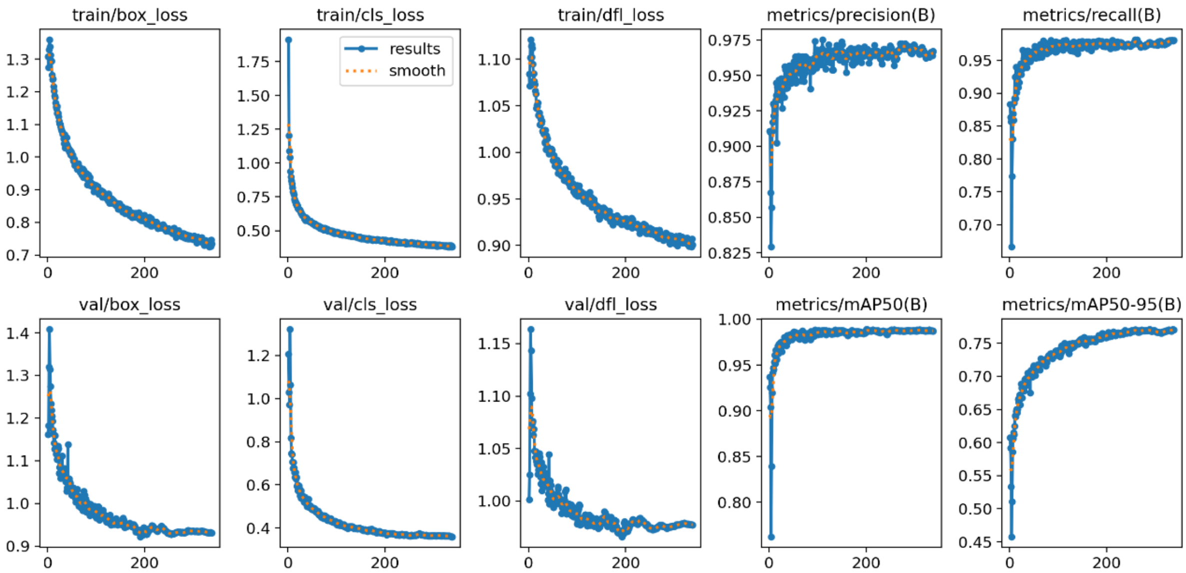


Figure 14. Human detection Model Training chart

Below are the technical comments for each graph related to the human detection model:

<u>Evaluation of Training Losses</u>

- *train/box_loss:* The loss for object boxes started at approximately 1.3 and decreased to about 0.8 after 200 epochs. This indicates that the model has learned to position objects more accurately.
- *train/cls_loss:* The classification loss started at around 1.75 but decreased to 0.5 as training progressed. This reduction shows that the model is better at correctly identifying the class labels of the objects.
- *train/dfl_loss:* The distribution loss began at a level of 1.1 and fell to about 0.9 after 200 epochs. This decrease suggests that the model has reduced uncertainty in its predictions and can make more precise predictions.

<u>Evaluation of Validation Losses</u>

- *val/box_loss:* The loss for object boxes on the validation data started at 1.4 and gradually decreased to approximately 0.9 after 200 epochs. This demonstrates that the model can also make accurate box placements on new data.
- *val/cls_loss:* The classification loss on the validation data shown a similar decrease. Initially at 1.2, it dropped to about 0.4 after 200 epochs.
- *val/dfl_loss:* The distribution loss started at 1.15 and fell to 0.95 by the end of the training process. This indicates that the model can also make more accurate predictions on validation data.

<u>Metrics for RT Orientation Detection Model</u>

- *metrics/precision(B):* The precision value, which was low at the beginning of the training process, rapidly increased to about 0.96 and stabilized. This indicates that the model is highly successful in making true positive predictions with very few false positives.
- *metrics/recall(B):* The recall value, initially low, also rose quickly to approximately 0.95 and stabilized. This shows that the model can detect true positives at a high rate, with very few false negatives.
- *metrics/mAP50(B):* The mean Average Precision metric at 50 (IoU = 0.5) was initially low but quickly rose to about 0.98 and stabilized. This indicates that the model has high overall prediction accuracy.
- *metrics/mAP50-95(B):* The mean Average Precision metric from 50 to 95 (IoU = 0.5 to 0.95) started low but rapidly increased to approximately 0.77 and stabilized. This indicates that the model demonstrates high prediction accuracy across different threshold values.

4.1.2. RT detection model performance

The model developed for RT detection was designed to detect RT vehicles and also their gaze directions. The model can accurately determine the given images using classification algorithms. The performances of the model developed for RT detection, which produces high accuracy results, have been examined in the following graphs.

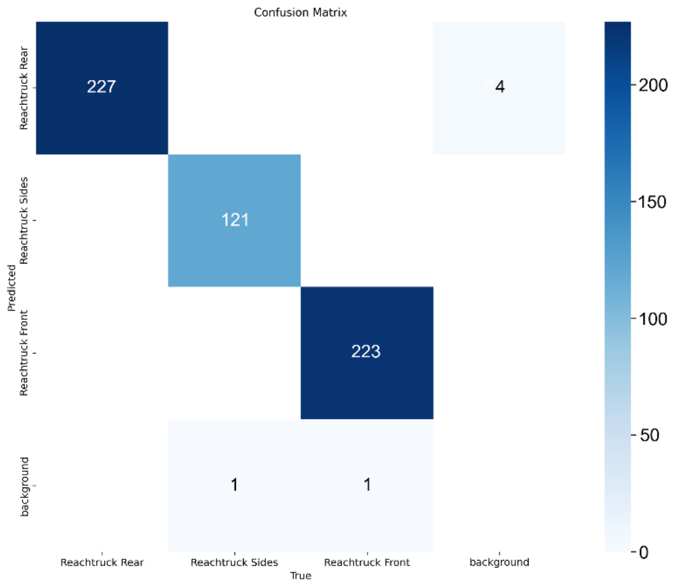


Figure 15. Confusion Matrix of the RT detection model

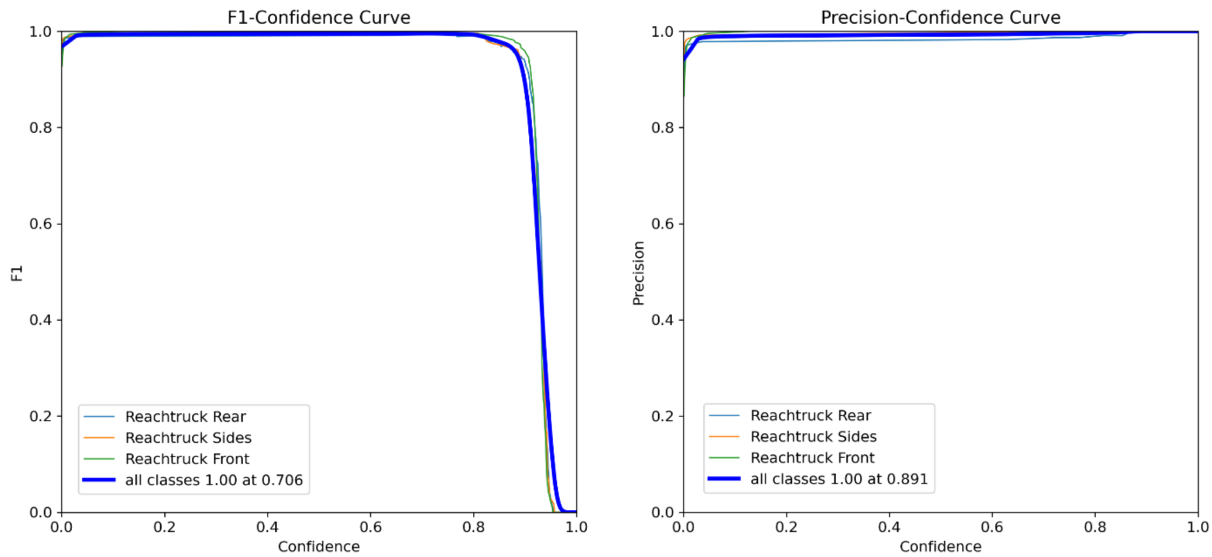
The confusion matrix in Figure 15 illustrates the performance of a four-class classification model. These classes have been divided into "Reachtruck Rear," "Reachtruck Sides," "Reachtruck Front," and "background." For the "Reachtruck Rear" class, the model correctly predicted this class 227 times (top-left cell) with no false negatives (FN) for this class. In the "Reachtruck Sides" class, the model made 121 correct predictions (second cell in the bottom-left corner), with only 4 incorrect predictions, indicating overall accurate classification for this class. For the "Reachtruck Front" class, the model correctly predicted this class 223 times (bottom-right corner) and also has no FN in this class. In the "background" class, the model correctly identified the "background" class only once (top-right cell), with no false positives (FP) for this class.

The high True Positive (TP) counts and the absence of FN in the "Reachtruck Rear," "Reachtruck Sides," and "Reachtruck Front" classes demonstrate the model's strong performance in recognizing these classes.

Since detecting the "background" class is not the primary goal of the project, the low performance in this class does not negatively impact the model's overall success. This evaluation highlights the model's high performance in the targeted Reachtruck classes, with lower success in the background class.

During model training, only 1% of the data used was from the "background" class. This proportion was strategically chosen to minimize visual confusion between Reachtruck and similar objects. While using more background data could improve the model's accuracy in identifying the background, it could also increase the risk of overfitting, thereby reducing accuracy. Thus, the 1% background data usage was deemed optimal for performance based on experiments. Background data was included specifically to prevent Battery Powered Pallet Trucks in the field from being misclassified as Reachtruck classes. Since Battery Powered Pallet Trucks and Reachtrucks can visually resemble each other, adding background data aimed to enhance the model's ability to distinguish between these objects and reduce misclassifications. This approach has improved the model's object detection accuracy and minimized classification confusion.

In Figure 16(a), the statement "all classes 1 at 0.706" in the f1 score graph indicates that the all f1 scores for all classes were 100% when a threshold value of 0.706 was used for RT direction detection. This suggests that the model is able to successfully f1 all classes even at a very high threshold value. Additionally, the statement "all classes 1 at 0.891" in the precision score graph in Figure 16(b) indicates that the precision scores for all classes are 100% when a threshold value of 0.891 is used. This demonstrates that the model has an excellent ability to correctly classify all classes when decisions have made above the 0.891 threshold.



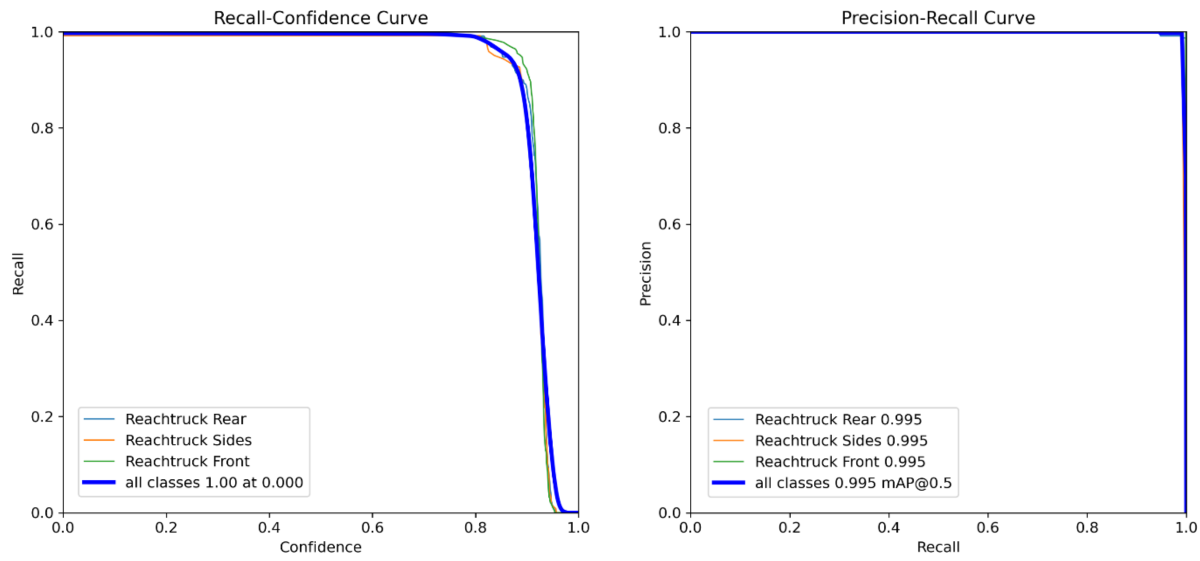
(a) (b)

Figure 16. F1 and Precision-Confidence score graph of the RT detection model

In Figure 17(a), the statement 'all classes 1 at 0.000' in the recall score graph indicates that the recall scores for all classes were 100% when a threshold value of 0 was used. This suggests that the model is able to successfully recall all classes even at a very low threshold value.

Additionally, the statement "all classes 0.995 at mAP@50" in the precision-recall score graph in Figure 17(b) indicates that the mAP scores for all classes are 99%. This suggests that the model is able to successfully recall and precission all classes even at a very low threshold value.

In Figure 18, it displays the losses and performance metrics of the object detection model during the training and validation process. Each of these graphs allows us to analyze a specific aspect of the model and its performance in detail.



(a) (b)

Figure 17. Recall and Precision graph of RT detection model

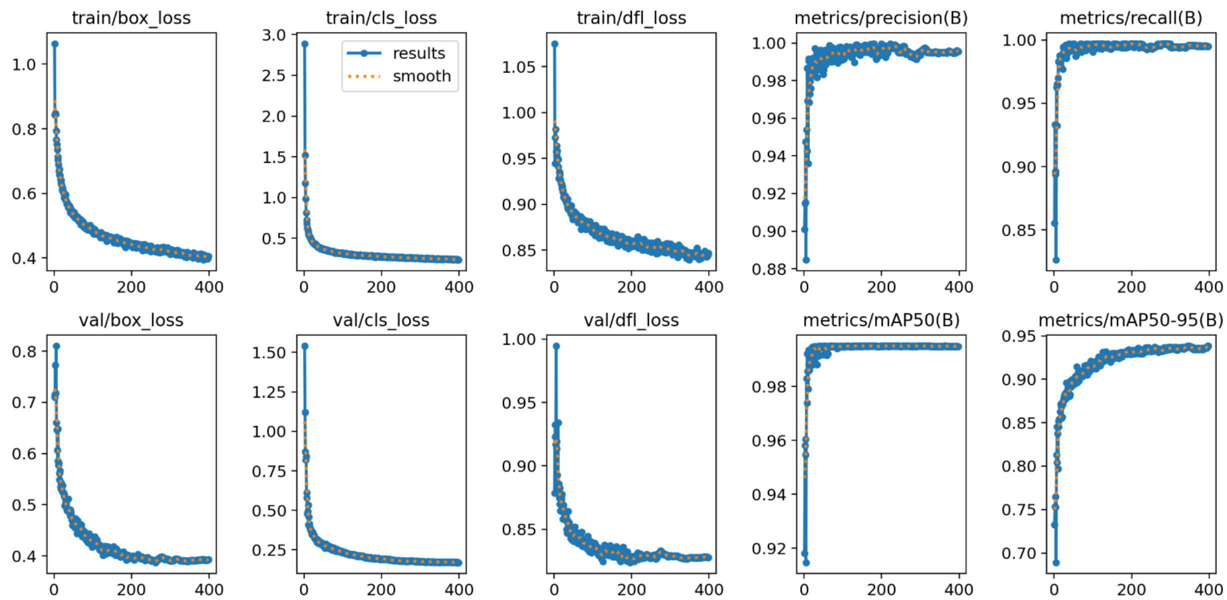


Figure 18. RT detection model training graph

Below are the technical comments for each graph related to the RT direction detection model:

Evaluation of Training Losses

- *train/box_loss:* This loss value, which represents the error in the positioning of object boxes during training, initially starts around 1.0 and decreases to approximately 0.4 after 200 epochs. This decrease indicates that the model has been learning to position objects more accurately.
- *train/cls_loss:* The classification loss starts at around 3.0 and gradually drops to 0.5 as training progresses. This reduction shows that the model has been learning to identify the class labels of objects more accurately.
- *train/dfl_loss:* The distribution loss starts at 1.05 and decreases to around 0.85 after 200 epochs. This reduction indicates that the model has been reducing uncertainty in its predictions and can make more precise predictions.

Evaluation of Validation Losses

- *val/box_loss:* The box loss on validation data starts at 0.8 and gradually drops to approximately 0.4 after 200 epochs, showing that the model can make accurate box placements on new data as well.
- *val/cls_loss:* Similar to the training data, the classification loss on validation data decreases from 1.5 initially to approximately 0.25 after 200 epochs.
- *val/dfl_loss:* The distribution loss starts at 0.99 and decreases to 0.85 by the end of training, showing that the model is also making more precise predictions on the validation data.

Metrics for RT Orientation Detection Model

- *metrics/precision(B):* Initially low, the precision value rapidly increases to approximately 1.0 and stabilizes. This shows that the model is highly successful in making correct positive predictions with very few false positives.
- *metrics/recall(B):* The recall value, which is initially low, rapidly rises to about 1.0 and stabilizes, demonstrating that the model can detect true positives at a high rate with very few false negatives.
- *metrics/mAP50(B):* The mean accuracy metric @ 50 (IoU = 0.5) starts low and quickly rises to about 1.0, indicating high overall prediction accuracy by the model.
- *metrics/mAP50–95(B):* The mean accuracy metric @ 50–95 (with IoU thresholds from 0.5 to 0.95) also starts low, increases rapidly, and stabilizes around 0.95, indicating that the model shows high prediction accuracy across various threshold values.

The model has also achived successful results on the images from different enviroment conditions and various aspects. Through to these features, it has the potential to be used as a reliable solution in automated RT recognition systems.

4.2. Social Navigation Tests

The developed social navigation system has been integreted into the PIXER robot platform. The system has been tested by the warehouse. The PIXER has been assigned to areas within an aisle width of 2.5 m and an approximate aisle length of 9 m. In the first situation, PIXER has been provided autonomous navigation on an empty way from A point to B point and the arrival time hes been measured as t_1=10,6 seconds. The movement of the robot was monitored in real-time via the PIXER tracking and control interface. The real-time status image of PIXER, recorded from the interface, is shown in Figure 19.

Figure 19. In an empty aisle, with the RT detection model deactivated, the process of the PIXER robot planning its route globally and initiating movement locally.

In the second scenario, the RT detection module has been activated on PIXER in the same aisle, and the robot has been instructed to move from point A to point B. Initially, PIXER has planned its route globally, then has proceeded to plan the local route and has started moving. This situation has been illustrated in Figure 20. However, upon detecting an RT along the route, PIXER has planned an alternative path. As a result, PIXER, having reached point B from point A, has been measured to have a travel time of t_2=30,6 seconds. PIXER's alternative route planning upon encountering an RT has been shown in Figure 21.

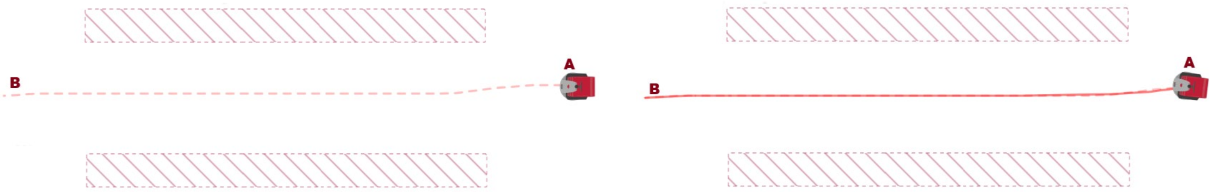


Figure 20. The process of the PIXER robot planning its route globally and initiating movement locally with the RT detection model activated.

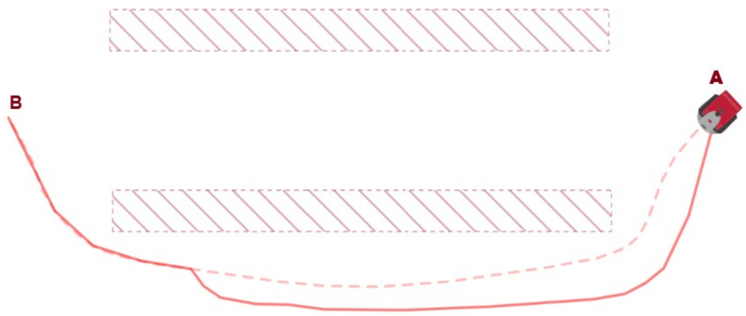


Figure 21. The PIXER robot's replanning of an alternative route upon detecting an RT along its path.

In the third scenario, the RT detection module on PIXER was deactivated in the same aisle, and the robot was instructed to move from point A to point B. Since the RT detection module was inactive on PIXER, it attempted to pass by the RT upon reaching its location and waited there. Due to this delay, PIXER eventually reached point B with a travel time of t_3=49,5 seconds. This situation is illustrated in Figure 22. A total of 5 different trials were conducted under 3 different conditions in the same environment. The time measurements obtained from these trials are presented in Table 3.

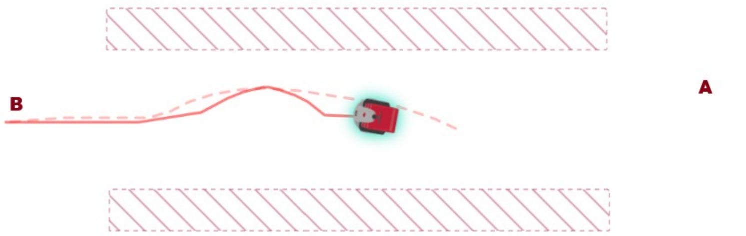


Figure 22. With the RT detection model active, the PIXER robot perceives the RT along its path as a regular object and attempts to pass by it.

Table 3. Studies in which the RT detection module is active and deactive

Test	Arrival time on empty road (sec)	Arrival time (sec) when RT detection model is active on the path where RT is located	Arrival time (sec) when RT detection model is disabled on the path where RT is present
1	10,6	30,6	49,5
2	10,9	32,2	50,3
3	9,5	29,5	49,8
4	10,8	31,8	50,7
5	11,2	33,4	51,2
Average	10,6	31,5	50,3

Some tests have been conducted by activating and deactivating the human detection model on PIXER. Similar to the RT detection model, PIXER has been instructed to move from point A to point B with the human detection model active and inactive under the same aisle conditions (Figure 23). The results of these tests have been presented in Table 4. For example, looking at the first row of the table: PIXER has reached point B from point A in t_1=10,8 seconds on an empty path. When the human detection model has been active and there has been a person along the route, PIXER has reached point B in t_2=14,5 seconds. When the human detection model has been inactive and there has been a person along the route, PIXER has reached point B in t_2=15,3 seconds. The results of 5 different trials conducted in this manner have been presented in the table. Additionally, a representative image of human detection from the depth camera on PIXER has been shown in Figure 24.

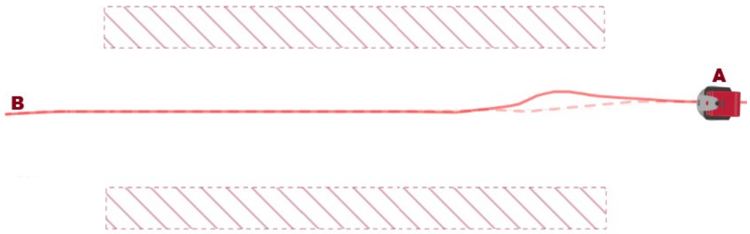


Figure 23. The scenario where a human is present during navigation in the aisle.

Figure 24. The scenario where a human is present during navigation in the aisle.

Table 4. Studies in which the human detection module is active and deactive

Test	Arrival time on empty road (sec)	Arrival time (sec) when the human detection model is active on the road with people	Arrival time (sec) when the human detection model is disabled on the road with people
1	10,8	14,5	15,3
2	10	15,2	15,9
3	9,9	15,6	14,6
4	10,7	14,9	14,8
5	11,2	16,1	16,2
Average	10,52	15,26	15,36

5. CONCLUSION

This study has comprehensively evaluated the social navigation performance of the PIXER robot in e-commerce warehouses, focusing on human and RT detection using YOLOv8n and depth cameras. Through the experiments conducted, the impact of active detection models on the robot's efficiency and safety has been examined during navigation in a 9-meter-long and 2.5-meter-wide aisle. In scenarios without any obstacles, the PIXER robot has successfully reached point B from point A in an average of 10.6 seconds. When the RT detection model has been active and an RT has been detected, the robot has replanned its path and reached its target in an average of 31.5 seconds. In contrast, without an active RT detection model, the robot has been unable to detect the RT and has had to wait to bypass the obstacle, reaching its target in 50.3 seconds. This situation has been shown in Figure 25. Similarly, the human detection model has also played a critical role in ensuring safe and efficient navigation. When the human detection model has been active, the robot, upon detecting a human, has reduced its maximum speed by 80% to pass at a safe distance, reaching its target in an average of 15.26 seconds. Without an active human detection model, however, the robot has failed to detect the human presence, faced potential collision risks, and experienced significant delays, reaching its target in an average of 15.36 seconds. This situation has been illustrated in Figure 26.

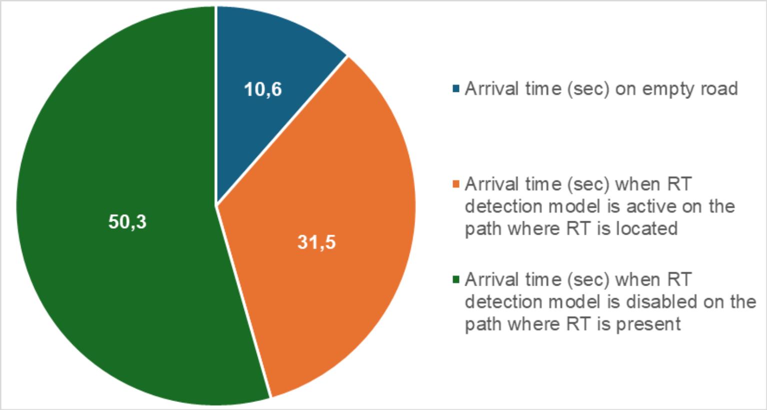


Figure 25. Average target arrival times (sec.) with the RT detection module active and inactive.

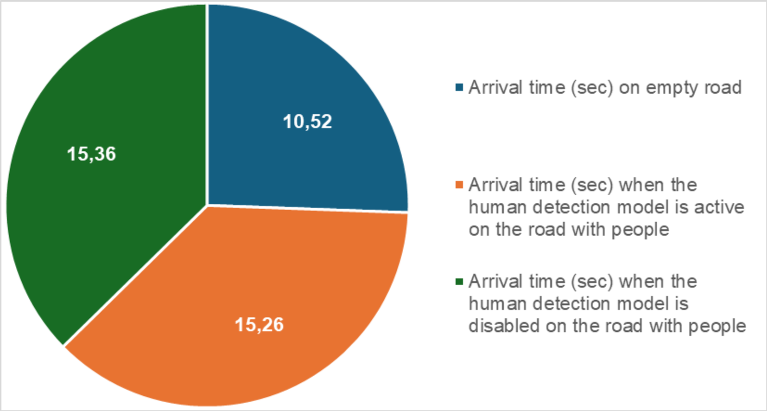


Figure 26. Average target arrival times (sec.) with the human detection module active and inactive.

When examining scenarios where the human detection model is active and inactive, it has been observed that the robot reached its target in approximately similar times in both cases. With the human detection model inactive, the robot behaved as if the obstacle were an ordinary dynamic object, slowing down but passing by in an unsafe manner without regard for what the object was. Additionally, because it hadn't pre-adjusted its speed, the robot approached the object too closely. However, in the case where the model was active, the robot detected the human and slowed down safely to avoid a collision. By adjusting its speed in advance, the robot avoided approaching the human object unsafely. These findings underscore the importance of integrating detection models to enhance the operational efficiency and safety of AMRs in dynamic environments. The notable time savings and reduction in collision risk demonstrate the value of advanced detection systems in supporting real-time decision-making and adaptive path planning processes.

Similar studies in literature primarily focus on areas such as human detection and maintaining a safe distance from humans. Unlike these studies, this research enhances the sensitivity of actively navigating AMR in a warehouse environment to dynamically moving objects in its surroundings. As a result, the AMR operates more efficiently in terms of both occupational safety and operational time.

This study was tested in a small-scale warehouse environment with a single AMR. Environments where multiple AMRs operate in larger warehouse settings are considered a potential topic for future research. Upcoming studies will focus on further optimizing detection algorithms to reduce latency and improve accuracy. Additionally, plans include expanding experimental setups to incorporate more complex and diverse warehouse environments. The exploration of multi-sensor fusion techniques could provide a more holistic perception framework, further enhancing the robot's navigation capabilities. Unlike this study, which focused on human and RT detection in social navigation, future work aims to develop new detection models for environments with other objects, such as electric pallet jacks.

Author Contributions
Bilal Gürevin: Literature review, Methodology, Analysis, Writing-original draft *Hilal Öztemel*: Modelling, Writing-review and editing *Burhan Turgut Ulutürk*: Literature review, Data Curation, Analysis *Emre Sebat*: Modelling, Writing-review and editing

Conflict of Interest
No potential conflict of interest was declared by the authors.

Funding
This study was supported by TUBITAK within the scope of “Project No: 3231501”.

Compliance with Ethical Standards
It was declared by the authors that the tools and methods used in the study do not require the permission of the Ethics Committee.

Ethical Statement
It was declared by the authors that scientific and ethical principles have been followed in this study and all the sources used have been properly cited.

REFERENCES

Beyer, L., Hermans, A., Linder, T., Arras, K.O. and Leibe, B. (2018). “Deep Person Detection in 2D Range Data”, arXiv preprint:1804.02463.

Clavero, C., Patricio, M.A., García, J. and Molina, J.M. (2024). “DMZoomNet: Improving Object Detection Using Distance Information in Intralogistics Environments”, *IEEE Transactions on Industrial Informatics*, 20 (7), 9163-9171. https://doi.org/10.1109/TII.2024.3381795

Cognominal, M., Patronymic, K. and Wańkowicz, A. (2021). “Evolving Field of Autonomous Mobile Robotics: Technological Advances and Applications”, *Fusion of Multidisciplinary Research An International Journal*, 2(2), 189-200.

Daza, M., Barrios-Aranibar, D., Diaz-Amado, J., Cardinale, Y. and Vilasboas, J. (2021). “An Approach of Social Navigation Based on Proxemics for Crowded Environments of Humans and Robots”, *Micromachines*, 12(2), 193. https://doi.org/10.3390/mi12020193

Fragapane, G., Ivanov, D., Peron, M., Sgarbossa, F. and Strandhagen, J.O. (2022). “Increasing Flexibility and Productivity in Industry 4.0 Production Networks with Autonomous Mobile Robots and Smart Intralogistics”, *Annals of Operations Research*, 308(1), 125-143. https://doi.org/10.1007/s10479-020-03526-7

Francis, A., Pérez-d'Arpino, C., Li, C., Xia, F., Alahi, A., Alami, R. and Martín-Martín, R. (2023). “Principles and Guidelines for Evaluating Social Robot Navigation Algorithms”, arXiv preprint arXiv:2306.16740.

Gürevin, B., Gül, R., Eğri, S., Gültürk, F., Yıldız, M., Çalışkan, F. ve Pehlivan, İ. (2023). “A Novel Method Determining the Size and Angle of an Object Using a Depth Camera Without Reference”, *Academic Platform Journal of Engineering and Smart Systems*, 11(2), 41-46. https://doi.org/10.21541/apjess.1297168

Gürevin, B., Yıldız, M., Gültürk, F., Pehlivan, İ., Çalışkan, F., Boru, B. ve Yıldız, M.Z. (2024). “A Novel Control and Monitoring Interface Design for ROS Based Mobile Robots”, *Düzce Üniversitesi Bilim ve Teknoloji Dergisi*, 12(1), 496-509. https://doi.org/10.29130/dubited.1214278

Intel Realsense. (2024). https://www.intelrealsense.com/depth-camera-d455/, (Erişim Tarihi: 25.07.2024).

Jia, D., Hermans, A. and Leibe, B. (2020). “DR-SPAAM: A Spatial-Attention and Auto-Regressive Model for Person Detection in 2D Range Data”, *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 10270-10277.

Kenk, M.A., Hassaballah, M. and Brethé, J.F. (2019). “Human-Aware Robot Navigation in Logistics Warehouses”, *Proceedings of the 16th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2019)*, 371-378.

Kim, C. E., Oghaz, M.M.D., Fajtl, J., Argyriou, V. and Remagnino, P. (2018). “A Comparison of Embedded Deep Learning Methods for Person Detection”, arXiv preprint arXiv:1812.03451.

Mavrogiannis, C., Baldini, F., Wang, A., Zhao, D., Trautman, P., Steinfeld, A. and Oh, J. (2023). “Core Challenges of Social Robot Navigation: A Survey”, *ACM Transactions on Human-Robot Interaction*, 12(3), 1-39. https://doi.org/10.1145/3583741

Mayershofer, C., Holm, D.M., Molter, B. and Fottner, J. (2020). “Loco: Logistics Objects in Context”, *19th IEEE International Conference on Machine Learning and Applications (ICMLA),* 612-617.

Önal, O. and Dandıl, E. (2021). “Object Detection for Safe Working Environments Using YOLOv4 Deep Learning Model”, *Avrupa Bilim ve Teknoloji Dergisi*, (26), 343-351. https://doi.org/10.31590/ejosat.951733

Reis, D., Kupec, J., Hong, J. and Daoudi, A. (2023). “Real-Time Flying Object Detection with YOLOv8”, arXiv preprint arXiv:2305.09972.

Salmerón-Garci, J., Inigo-Blasco, P., Di, F. and Cagigas-Muniz, D. (2015). “A Tradeoff Analysis of a Cloud-Based Robot Navigation Assistant Using Stereo Image Processing”, *IEEE Transactions on Automation Science and Engineering*, 12(2), 444-454. https://doi.org/10.1109/TASE.2015.2403593

Trakadas, P., Simoens, P., Gkonis, P., Sarakis, L., Angelopoulos, A., Ramallo-González, A. P., Skarmeta, A., Trochoutsos, C., Calvo, D., Pariente, T., Chintamani, K., Fernandez, I., Irigaray, A.A., Parreira, J.X., Petrali, P., Leligou, N. and Karkazis, P. (2020). “An Artificial Intelligence-Based Collaboration Approach in Industrial IoT Manufacturing: Key Concepts, Architectural Extensions and Potential Applications”, *Sensors*, 20(19), 5480. https://doi.org/10.3390/s20195480

Truong, X.T. and Ngo, T.D. (2016). “Dynamic Social Zone Based Mobile Robot Navigation for Human Comfortable Safety in Social Environments”, *International Journal of Social Robotics*, 8, 663-684. https://doi.org/10.1007/s12369-016-0352-0

Ultralytics. (2024a). https://docs.ultralytics.com/tr/models/yolov8/, (Erişim Tarihi: 25.07.2024).

Ultralytics. (2024b). https://github.com/ultralytics/ultralytics/tree/main, (Erişim tarihi: 08.10.2024).

Wang, F. and Li, X. (2023). “Research on Warehouse Object Detection for Mobile Robot Based on YOLOv5”, *2023 International Conference on Image Processing, Computer Vision and Machine Learning (ICICML)*, 1196-1200.

Zhu, K. and Zhang, T. (2021). “Deep Reinforcement Learning Based Mobile Robot Navigation: A Review”, *Tsinghua Science and Technology*, 26(5), 674-691.